A4 Cont B3

(d) means to compare said memory commands of one of said types with a current chosen memory command of the same type to determine which of said memory commands have the least memory cycle performance penalty;

(e) means to compare said memory commands of one of said types with a previously chosen memory command of the same type determine which of said memory commands have the memory cycle performance penalty;

(f) means to select one of said memory commands having the least memory cycle performance penalty by selecting the oldest; and

(g) means to continue execution of memory commands of the same type as said selected memory command.

REMARKS

The application was filed on 10 September 1999 with twenty-one claims. In the first Examiner's Office Action mailed 03 October 2002, the Examiner rejected claims 4-7, 17, 16 and 19, under 35 U.S.C. §112; the Examiner further rejected claims 19 and 20 under 35 U.S.C. §102(e) as being anticipated by U.S. Patent 6,385,708 entitled, "Using a Timing-Look-Up-Table and Page Timers to Determine the time Between Two Consecutive Memory Accesses," to Stracovsky et al. (Stracovsky '708). The Examiner further rejected claims Under 35 U.S.C. §103(a) as being unpatentable over Stracovsky '708 in view of U.S. Patent No.

6,088,772 entitled, "Method and Apparatus for Improving System Performance when Reordering Commands," to Harriman et al. (Harriman '772). The Examiner further objected to the specification and requested that it be revised. In response, Applicants corrected three minor typographical errors within the specification. Applicants further amended claims 1, 16, 17, and 19 to remove the rejections under 35 U.S.C. §112. With respect to the rejections of claims 19 and 20 under 35 U.S.C. §102(e) in light of Stracovsky '708 and with respect to the rejection of claims 1-21 under 35 U.S.C. §103(a) in view of Stracovsky '708 and Harriman '772, Applicants respectfully traverse without amendment.

Applicants submit formal drawings herewith.

The Specification

Applicants have responded to the Examiner's request to revise the specification. Applicants have corrected errors of the specification and now believe that the specification complies with 35 U.S.C. §112, first paragraph.

The Rejection Under 35 U.S.C. §112, second paragraph

Applicants have amended claim 1 to provide an antecedent basis for the "bus networks" of claims 4-7.

Applicants have amended claim 17 to replace "said determination" with the physical location and address dependency, if any, corresponding to said memory command [passed] along with said memory command. Thus, not "all" the dependencies are passed; only those dependencies associated with a particular memory command.

Applicants have further amended claim 16 to insert the word "until", i.e., now the step of executing continues "until" a valid command is not available. One of skill in the art will understand how execution continues.

Applicants further amend claim 19; first to insert a comma to indicate that the phrase "said plurality of bus units interconnected on a bus network" is separate from the phrase "said plurality of bus units to issue memory commands". Applicants have further incorporated the Examiner's suggestion to move the phrase "on a first bus" to modify the verb connected to .... The Applicants thank the Examiner for his careful reading of the specification and of the claims.

The Rejection Under 35 U.S.C. §102(e)

The Examiner rejected claims 19 and 20 under 35 U.S.C. §102(e) alleging that Stracovsky '708 anticipates the claimed invention. Applicants respectfully traverse the rejection because Stracovsky '708 does not disclose "a plurality of comparison logic circuits, each of said plurality of comparison logic circuits associated with each of said plurality of command FIFO queues to determine which memory commands of each of said command types **have the least memory cycle performance penalty**.

Stracovsky '708 reveals a memory system designed to avoid collisions. Applicants teach a memory system designed to execute those memory commands having the least memory latency. In the art of memory controllers and subsystems, these are two very different problems. The Examiner cites Stracovsky '708 as having a comparison logic circuit and references column 20, lines 11-30 of the reference. The purpose of the comparator matrix 1506 to which the Examiner refers is to synchronize the arrival of data with its memory command that has a command issue time, C_d, of zero, i.e., the command is ready to issue. After the command issue time of a command is zero, the ...

> [c]omparator matrix 1506 performs a **collision detection** function in which the data occurrence time of a command ready to be issued from command queue is compared to the data occurrence times of **previously issued**

> commands, as represented in data queue 1504. If a collision is detected, issuance of the command is delayed. Stracovsky '708, column 18, lines 56-63. (Emphasis added)

Moreover, these commands that are compared correspond to the same logical bank in memory (column 19, lines 7-8).

Applicants have claimed a very different invention. Applicants compare the memory latency of commands to be issued, not to avoid data collision between a command to be issued and other commands that have already issued. Applicants claim an invention to allow those commands with least memory latency to issue regardless of whether the commands are to the same logical bank in memory for the purpose of keeping the bus and memory controller busy, not to avoid collisions to the same memory bank.

Thus, Applicants respectfully traverse the rejection of the claims as being anticipated under 35 U.S.C. §102(e) by Stracovsky '708 because Applicants do not compare a memory command to be issued with previously issued commands to the same memory bank for the purpose of avoiding collisions. Rather, Applicants respectfully have claimed a method and an apparatus that compares the memory latency of a plurality of commands to be issued and then, depending of the latency, can reorder the execution of the commands.

The Rejection Under 35 U.S.C. §103(a)

The Examiner has further rejected the claims as being obvious in view of Stracovsky '708 in view of Harriman '722. Stracovsky '708, as explained, proposes a memory controller having a circuit to avoid data transfer collision of memory commands to the same memory bank. Harriman '722 teaches a memory access controller in which memory commands to the same memory portion are first collected and then commands to the same memory portion are continued to execute until a count is reached. It would appear, that Stracovsky

'708 and Harriman '722 should not be combined, and even if they were combined, they still would not teach the apparatus and method of selecting commands based on the least memory latency. First, Stracovsky '708 teaches a method to avoid collisions by calculating the data occurrence time in which data is expected to arrive from a memory portion accessed by five previously executed commands. If a new command ready to issue to that same memory portion is expected to collide with data already on the bus from the first memory command, then the new command must wait until after the data occurrence time. Harriman '722, on the other hand, teaches that it is more efficient to access the same memory structure to allow the row address strobe (RAS) to remain active. While it is possible to envision how Harriman '722 might be combined with Stracovsky '708, the combination is not suggested by either reference because one teaches the inefficiency resulting from the possibility of collisions when accessing the same memory bank while the other teaches that it is more efficient to access the same memory bank in order the keep the row bits active. More importantly, however, the combination of references do not teach Applicants' claimed invention of selecting the memory commands having the least memory latency with an entire memory subsystem, not such to a single memory portion. Indeed, Harriman '722 teaches against Applicants claimed invention, because Applicants in the originally filed specification, at page 17, lines 1-20, that the memory penalty of accessing the same memory bank, the precharge penalty, is greater than the memory penalty accessing a different memory bank or card, the switching penalty. Thus, one of skill in the art would not be inclined to consider either Stracovskly '708 but especially, not Harriman '722 when comparing memory cycle penalties.

Conclusion

Applicants have amended the specification and the claims to remove the objections and rejections based on 35 U.S.C. §112, first and second paragraphs. Applicants respectfully traverse the rejection of anticipation based on Stracovsky '708 because it does not teach comparing the memory penalties of commands yet to be issued. Applicants respectfully traverse the rejection of obviousness based on Stracovsky '708 and Harriman '722 because Harriman '722 teaches a method to access the same the memory bank and Applicants teach that accessing the same memory bank would impose the largest memory cycle penalty. Applicants respectfully request the Examiner to reconsider the application in view the amendments and the remarks, and respectfully request the Examiner to pass the application to issuance. The Examiner is further invited to telephone the Attorney listed below if he thinks it would expedite the prosecution and the issuance of the patent.

Respectfully submitted,
Herman L. Blackmon et al.

Date: By Karuna Ojanen

Karuna Ojanen
Registration No. 32,484
(507) 285-9003 voice
(507) 252-5345 fax

IBM Corporation
Intellectual Property Law Dept. 917
3605 Highway 52 North
Rochester, MN 55901-7829

APPENDIX

Amendments to the specification and claims showing deletions and additions:
U.S. Serial No. 09/394,011 filed 10 September 1999
Attorney Docket No. ROC919990080US1

The original paragraph on page 3, lines 8-20, with the amendment shown is:

Still other techniques to improve memory subsystem performance [inlcud] <u>include</u> overlapping and interleaving commands to the memory devices. Interleaving to route commands on different memory buses or different memory cards was improved by providing additional memory in the form of devices or more memory banks. But interleaving more devices or more banks requires more 1/0 pins, more power, and more cost to the entire system to interconnect the memory banks. The amount of data processed with each access to memory was increased to improve memory bus utilization. Similarly the data bus width could be narrowed, but a decreased bandwidth would decrease overall performance. Memory subsystem performance has improved as a result of all these improvements but it still remains a slower aspect of computer processing in which memory clocks typically operate two to four times slower than processor clocks.

The original paragraph on page 7, lines 8-19, with the amendment indicated is:

Another aspect of the invention is a computer memory controller, comprising means to receive a plurality of types of memory commands from a plurality of command sources; means to determine the memory cycle performance penalty associated with each memory command of each [of] type; means to compare the memory commands of one type with other memory commands of the same type, with a current chosen memory command of the same type, and with a previously chosen memory command of the same type to determine which of the memory commands have the least memory cycle performance penalty; means to select the oldest of the memory commands having the least memory cycle performance penalty; and means to continue execution of memory commands of the same type as the selected memory command.

The original paragraph of page 18, lines 1-16, with amendments, is:

command prioritizing and reordering aspects of the invention will be described. As discussed previously with respect to Figure 4, the commands in each of the FIFOs 510 are all of the same type. FIFO 510 in Figure 5 is shown as having sixteen entries wherein the bottom entry is the oldest and the command in entry sixteen is the most recently received command. A number of commands[,] which in the embodiment described herein [it] is the four oldest commands, [that] are selected for penalty comparison in blocks 520 and 530 of Figures 5 and 6. In block 520, the four oldest commands selected from the FIFO 510 are compared with the current command in the chosen command register 454 to determine the cycle penalty of the four oldest commands selected from the FIFO 510. In block 530, the four oldest commands selected from FIFO 510 are compared with a number of the previous chosen commands stored in the previous chosen command register 458, again to determine the cycle penalty of the four oldest commands. In block 540, the four oldest commands selected from FIFO 510 are compared amongst each other to determine which has the least cycle penalty and which is the oldest.

Claims 1, 16, 17, and 19 have been amended.

1. (Amended) A method to process commands in a computer memory subsystem, comprising:

   (a) receiving a plurality of commands on a bus network connected to said memory subsystem;

   (b) categorizing said received commands into command types;

   (c) determining memory cycle performance penalties of said categorized commands;

   (d) reordering said categorized commands so that said categorized commands having the least memory cycle performance penalty are selected for execution;

   (e) determining if said reordered commands are valid;

   (f) arbitrating said valid commands;

   (g) executing sequential valid commands of the same command type.

16. (Amended) The method of claim 1, wherein said step of executing sequential valid commands of the same command type further continues <u>until</u> a valid memory command of said command type is no longer available, or until a predetermined number has been executed, or until a memory command of another of said command types has higher priority.

17. (Amended) A method to process commands in a computer memory subsystem, comprising:

(a) receiving a plurality of memory commands on a bus connected to said computer memory subsystem and determining the physical location of the memory command in memory, and further determining if any of said received <u>memory</u> commands have an address dependency and passing said [determinations] <u>physical location and said address dependency, if any, corresponding to said received memory command</u> along with said memory command;

(b) categorizing said received commands into command types based on one of the following: STORE, FETCH, INTERVENTION STORE; the source or destination of said received memory commands; the program or application from which said memory commands originate or are otherwise required;

(c) determining memory cycle performance penalties of said categorized commands by comparing a number of oldest received categorized commands with each other, with a currently chosen command, and with a previously chosen command;

(d) reordering said categorized commands so that said categorized commands having the least memory cycle performance penalty are selected for execution and if more than one categorized command has the least memory cycle performance penalty, then selecting the oldest of said reordered commands for execution;

(e) determining if said reordered commands are valid;

(f) granting priority to said type of command having said least memory cycle performance penalty;

(g) executing sequential valid commands of the same command type until a valid command of the same type is not received or until a predetermined number has been executed, or until a memory command of another type has higher priority;

(h) avoiding deadlock when an address dependency exists between commands of different types by executing commands having the command type of the oldest memory command.

19. (Amended) A computer processing system, comprising:

(a) a plurality of bus units, said bus units comprising at least one computer processor, at least one I/O device; at least one memory cache system connected to said at least one computer processor, and at least one network communication device, said plurality of bus units interconnected on a bus network, and said plurality of bus units to issue memory commands, said memory commands categorized into types;

(b) at least one memory subsystem connected on a first bus to said plurality of bus units [on a first bus], said memory subsystem responsive to said memory commands and further comprising:

(i) a memory controller connected to a command interface functionally connected to said first bus;

(ii) a plurality of memory chips configured into memory banks; said memory chips architected into memory cards attached to at least one memory bus;

(iii) a plurality of command FIFO queues, each of said command FIFO queues associated with one of said command types into which said memory commands are categorized;

(iv) a plurality of comparison logic circuits, each of said plurality of comparison logic circuits associated with each of said

plurality of command FIFO queues to determine which memory commands of each of said command types have the least memory cycle performance penalty;

(v) an arbitration logic circuit to output said memory commands of said determined command type having said least memory cycle performance penalty to said plurality of memory chips.